An IoT-Enabled Deep Learning Framework for Crop Disease Detection Using Image Signal Processing and Convolutional Neural Networks.

Mr.Yadav Uttam Panaswad[1],
Prof. V. A. Daware[2]
[1]M.Tech Student, College of Engineering, Osmanabad.
[1]Assistant Professor, College of Engineering, Osmanabad.
Corresponding Author Email- uttampanaswad@gmail.com

Keywords

Crop Disease Detection
Deep Learning
Convolutional Neural Network
Internet of Things
Smart Agriculture
Plant Disease Classification

Abstract

The agriculture industry provides food and economic stability for the entire world population. Nevertheless, crop diseases still pose serious problems, resulting in considerable economic losses and decreasing crop yields. Therefore, timely and precise identification of the diseases becomes critical in minimizing their consequences. The conventional method used in detecting crop diseases is based on manual examination of crops, which requires extensive labor input, is not always precise, and depends heavily on the experts' knowledge. Such features make this approach impractical for modern farming.The present research introduces an innovative IoT-based approach for automated identification of crop diseases through image processing and CNNs. Within this system, sensors and smart cameras can be installed in crop fields to take and send photos of the crop leaves to the processing unit. Further, images are subjected to the preprocessing stage to reduce noise and normalize size and color values. The preprocessed images are further used by applying a CNN-based model, that is, automatic extraction of important attributes such as color changes, textures, disease-related characteristics, and more. Further, the proposed model is capable of classifying the images into different diseases with great accuracy. In addition to that, state-of-the-art techniques, such as attention mechanism and Grad-CAM, can be utilized for improvement in performance.The suggested system offers real-time monitoring capabilities as well as the results are provided via a user-friendly interface. As a result, the farmers are able to take quick and informed actions. With the reduction in dependency on manual inspection methods, as well as increased accuracy of image classification, the overall agricultural production process becomes much more efficient and productive.

1. Introduction

The agricultural sector plays an immensely crucial role in the economic development and food security of any nation. There are millions of people around the world who depend on agriculture as their source of earning money. Nevertheless, crop diseases are known to be extremely hazardous factors that lead to huge financial losses due to low productivity of crops. The reason behind these diseases may include fungi, bacteria, viruses, and environmental hazards. Hence, crop diseases need to be identified on time to avoid huge losses from agriculture [1].
In conventional techniques, crop disease detection can be performed manually where humans have to play a central role in the entire process. This method takes quite a lot of time since humans have to inspect each crop in the farm to identify the diseases. Moreover, when the number of crops increases, it will become impossible for humans to inspect the entire farm [2].
Through technological developments, computer vision and machine learning algorithms have been applied to automate the procedure of disease identification. These algorithms utilize images of leaves to detect any abnormalities in plants such as spotting, color change, and texture differences. In these algorithms, deep learning models like the Convolutional Neural Network (CNN) have proven to be very efficient at classifying images. The convolutional neural network can extract significant features from an image without requiring manual feature selection, thus enhancing its performance in plant disease detection [3], [4].
Over the last few years, the incorporation of the Internet of Things (IoT) has made great strides in automating agriculture. The Internet of Things comprises electronic devices that can be installed in agricultural farms to record real-time information about crops. Through these IoT devices, images of plants can be captured and transmitted for analysis in a processing unit [5].
Image preprocessing algorithms have proven valuable in boosting the efficiency of the disease identification system. The techniques include noise reduction, image improvement, image resizing, and segmentation. This helps to ensure that the images captured are of good quality and isolate the affected areas in the leaf. Consequently, the deep learning algorithm receives appropriate and relevant inputs, enabling it to provide accurate predictions.

However, despite the progress made, most existing approaches have failed to be applicable in real-world settings. The algorithms rely on a large amount of training data, which may not be available in many instances. Also, some of these algorithms require massive computing power, making it hard to implement them. Moreover, external factors like different lighting conditions affect the accuracy of such models when applied in practical situations..

To solve all these problems, this project suggests an IoT-integrated deep learning-based approach to disease detection for crops. This project uses IoT devices for real-time image acquisition, image signal processing for improving data quality, and convolutional neural networks for classifying the diseases of crops with a higher degree of accuracy.

A. Contribution of this Research :

1. Integration of IoT in Agriculture: This research aims at studying the application of IoT devices such as smart cameras and sensors in the process of obtaining real-time images and constant monitoring of agricultural fields.
2. Deep Learning Models: This research studies the application of convolutional neural networks (CNNs) to automatically detect crop diseases by examining images with a high level of accuracy.
3. Application of Image Signal Processing Approaches: This project examines the process of data preprocessing and data enhancement, which involves image enhancement and segmentation to improve image quality.
4. Identification of Research Gaps: The research highlights certain gaps that still exist in current researches, namely real-time detection inability, limited diversity of datasets, and lack of frameworks integrated with IoT

2. Related Work

Although there have been some remarkable improvements in the use of machine learning and deep learning algorithms in crop disease detection, there are still many obstacles that need to be overcome. These include reliance on high-quality data sets, computational difficulties, and inability to integrate the system into real-time applications.

The recent progress in Artificial Intelligence (AI) and Deep Learning (DL) has seen great strides in automated crop disease detection using leaf images. Nyawose et al. [1] provided an extensive overview of machine learning and deep learning algorithms. They found that CNNs outperformed conventional methods because they could automatically detect features. Nevertheless, their study failed to provide real-time IoT for field testing. In the same vein, Dhaka et al. [2] evaluated some CNN models like AlexNet, VGG, and ResNet. They showed great performance improvement but their model was highly dependent on large labeled data sets and did not allow real-time implementation.

Hasan et al. [3] explored various deep learning problems like data labeling issues, expensive computation, and scalability. This research indicated that semi-supervised learning strategies were used to resolve the issues mentioned above. Abade et al. [4] reviewed CNN-based methods and proved that the techniques were effective for image recognition; however, there were restrictions concerning the lack of implementation in real life and compatibility with IoT devices. Agriculture studies utilizing Mishra's approach like the work of Shelar et al. [5] introduced CNN-based approaches where accuracy was improved with the help of preprocessing strategies. Still, such algorithms have a limitation in terms of using small data and not implementing real-time monitoring.

Further developments include the utilization of the EfficientNet-based model by Sun et al. [6]; this model had higher accuracy because of proper data extraction and data augmentation techniques. However, the model requires a lot of computational costs and is applied mostly to laboratory datasets. According to Ahmad et al. [7], the significance of CNN became more evident after ImageNet achievement; the authors recommended combining CNN and IoT technologies..

However, Upadhyay et al. [8] discussed modern computer vision algorithms for detecting crop disease, emphasizing the need for real-time systems but mentioning high costs and scalability. Hukkeri et al. [9] designed CNN-based models that detect multiple diseases with higher accuracy; however, their model is not a real-time system and relies on the quality of datasets. Demilie et al. [10] compared machine learning and deep learning models and found that CNN offers more accurate outcomes, despite its computational complexity.

In recent times, researchers have proposed more advanced and hybrid models for detecting crop diseases. Capsule Networks [12], which offer better classification accuracy and deal with distortions, raise computational complexity. The hybrid models of CNN and LSTM [15], which consider both spatial and temporal attributes, offer better accuracy and prediction but take longer to train. Models based on transfer learning, such as EfficientNet [13], also deliver good results with small datasets.

In addition, although optimization-based techniques, such as BWOM [14], enhance accuracy, they lead to an increase in complexity. Comparative analyses of CNN architecture, including VGG, ResNet, and MobileNet, have established that lightweight models can be used in mobile applications, despite variations in accuracy levels. Real-time systems, combined with mobile applications [17], give quick results to farmers, although they rely heavily on internet and environmental conditions.

Examples of advanced systems, such as PDD-DL [18], prioritize scalability and real-time detection but rely on expensive hardware. On the other hand, hybrid ML-DL models enhance efficiency but lead to increased costs and complexity. A systematic review of literature [20] has established the existing gaps in current research, mainly lack of IoT-based models and real-time systems.

Innovations in recent years include attention-based CNNs [22], region-based CNNs [23], and multi-class classifiers [24]. While these models improve accuracy, they increase computational demands. Finally, integration of AI and IoT models [25] allows for constant monitoring and decision-making, although high costs and complex infrastructure make them unaffordable in small-scale farms..

Figure 3.1 provides examples of healthy and diseased structures in crop leaves, where differences include a regular pattern for healthy leaves and an irregular pattern for diseased leaves. Such differences form essential attributes that help in classifying the leaves using deep learning algorithms.

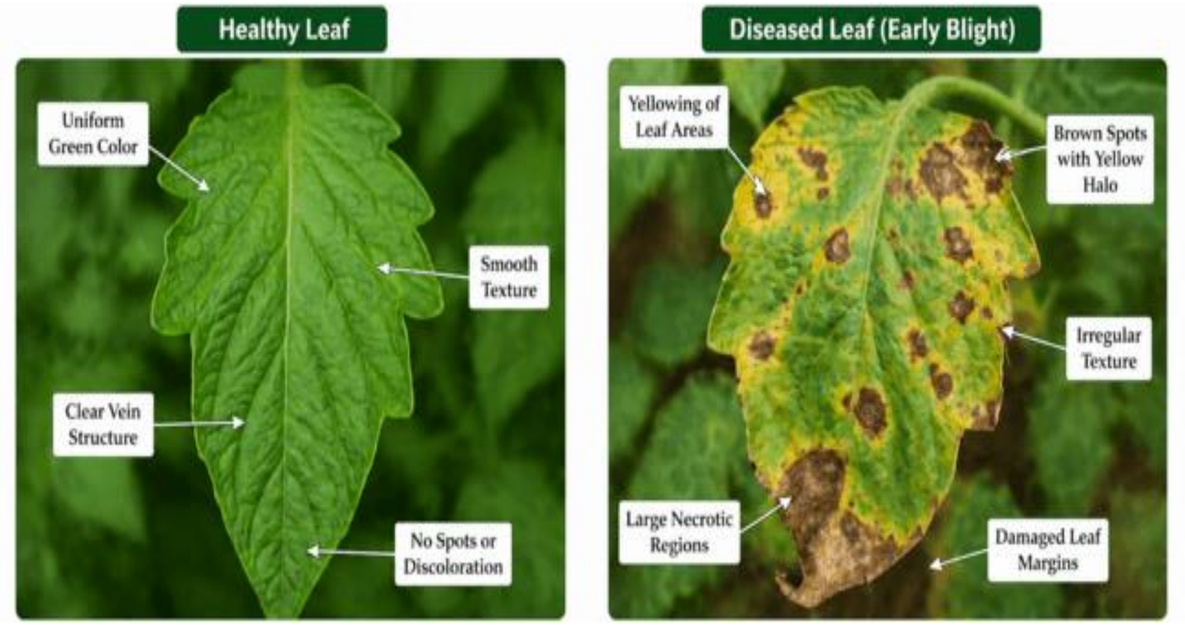

Fig. 1: Comparison of Healthy and Diseased Crop Leaves

From the above diagram, one can observe a stark contrast in terms of appearance between the healthy leaf structure and a diseased leaf suffering from early blight. From the healthy leaf structure, we notice that it has a consistent structure with a uniform green color, smooth texture, and well-defined veins, implying the plant is growing healthy and undergoing physiological processes without any complications. This further means that there are no spots or discolorations, which are signs of healthiness of the leaf and freedom from any form of infections. On the other hand, from the diseased leaf, we see various signs including yellowing of some areas, brown spots with yellow halos, and big dead areas. In addition, the texture of the leaves becomes irregular, with damaged edges. All these are some of the common signs of fungal infection such as early blight. All these features are key to the classification process. Through IoT-based cameras, images will be taken and processed before being classified using CNNs.

Table 1: Traditional Crop Disease Detection Methods and Their Limitations

Traditional Technique	Application	Limitation
Manual Inspection	Visual identification by farmers	Time-consuming and inaccurate
Laboratory Testing	Disease confirmation using chemical analysis	Expensive and slow
Image Processing (Traditional ML)	Feature-based classification	Requires manual feature extraction
CNN-based Models	Automatic disease detection	High computational cost
IoT-based Monitoring	Real-time data collection	High infrastructure cost

3. Methodology used for detection of Crop Detection

3.1 ResNet18

ResNet18 is one of the deep learning algorithms that can be used to classify images, and in this case, it will be used to confirm the presence of leaves in any given image. The primary function of ResNet18 is to overcome the problem of vanishing gradients in deep neural networks through the use of skip connections that enable data to move between layers without getting lost in the process. Instead of trying to learn a mapping function, it learns a residual function, making it easier to train. The algorithm has 18 layers, including convolutional layers and residual blocks, which aid in the extraction of relevant information from the images. This system uses the algorithm to perform a binary classification task (presence of leaves), thus ensuring that only images containing relevant leaves go through the subsequent processes. This makes the system more reliable and eliminates errors during disease prediction

3.2 Squeeze-and-Excitation (SE) Attention Mechanism

The SE (Squeeze and Excitation) Attention method can be applied for deep learning in order to emphasize the crucial features of an image, thus increasing classification performance. The primary purpose of the SE approach is to strengthen the capacity of a CNN to capture feature channels by applying specific weights to them. First, in the "squeeze" procedure, global information contained in a feature map is reduced to a small number of values that are then fed into the next layer of the network. The latter, known as "excitation," is responsible for evaluating the relevance of certain features, thus providing their specific weights. For instance, crucial disease spots or texture changes are highlighted, while insignificant background elements become almost irrelevant.

3.3 CNN-Bidirectional LSTM Hybrid Model

The CNN-Bidirectional LSTM architecture is an innovative combination of CNN and LSTM techniques that help increase the accuracy of disease detection. In essence, the primary purpose of this network is to identify both spatial and sequential dependencies within images of leaves. On one hand, the CNN model identifies visual features like edges, textures, and colors in images, while on the other hand, the LSTM model takes those features as inputs and analyzes their sequences for relationships within different parts of the leaf image. This approach, known as Bidirectional LSTM, enables the analysis of an image from two perspectives at once. Therefore, a hybrid model offers a better understanding of the structure of diseases within images compared to CNN alone

.

3.4 Grad-CAM (Gradient-weighted Class Activation Mapping)

Grad-CAM is one of the XAI methods used to visually interpret how the deep learning model generates its predictions. Its primary function is to emphasize the significant regions in an input image that impact the model's decision. It achieves this by computing the gradient of the predicted class score with respect to the activation map of the final convolutional layer in the neural network. The output is used to produce a heatmap that indicates which parts of the image are more important in generating the prediction. The heatmap is then superimposed over the image for easy interpretation of what the model sees as the essential parts of the image.

3.5 Canny Edge Detection and Image Fusion

Canny Edge Detection is an image processing technique that detects edges and structural information from an image. Canny Edge Detection is used to emphasize significant edges like leaf veins and diseases present in the image. This includes a number of steps, including reducing noise, calculating gradients, and detecting edges using thresholding methods. Once the edges are detected, the edges are overlaid on the original image by the use of image fusion techniques. This gives a clearer view of the structural information and any diseases present in the image. With this approach, the user can easily see the area where the diseases are occurring in the leaf.

3.6 HSV Color Masking (Green Ratio Detection)

The HSV Color Masking process is an image pre-processing step that helps to guarantee that the provided image has a proper plant leaf. It can be described as a procedure used to filter unnecessary images based on the presence of a certain color (green). To begin with, the RGB color representation of the image should be converted into the HSV color space as the latter isolates color information from light information. Then, the green color mask is applied, and the number of green pixels in the image is counted relative to the total pixel number. Should the obtained ratio fall under the required level, the picture will be excluded from further analysis as it does not provide enough plant information.

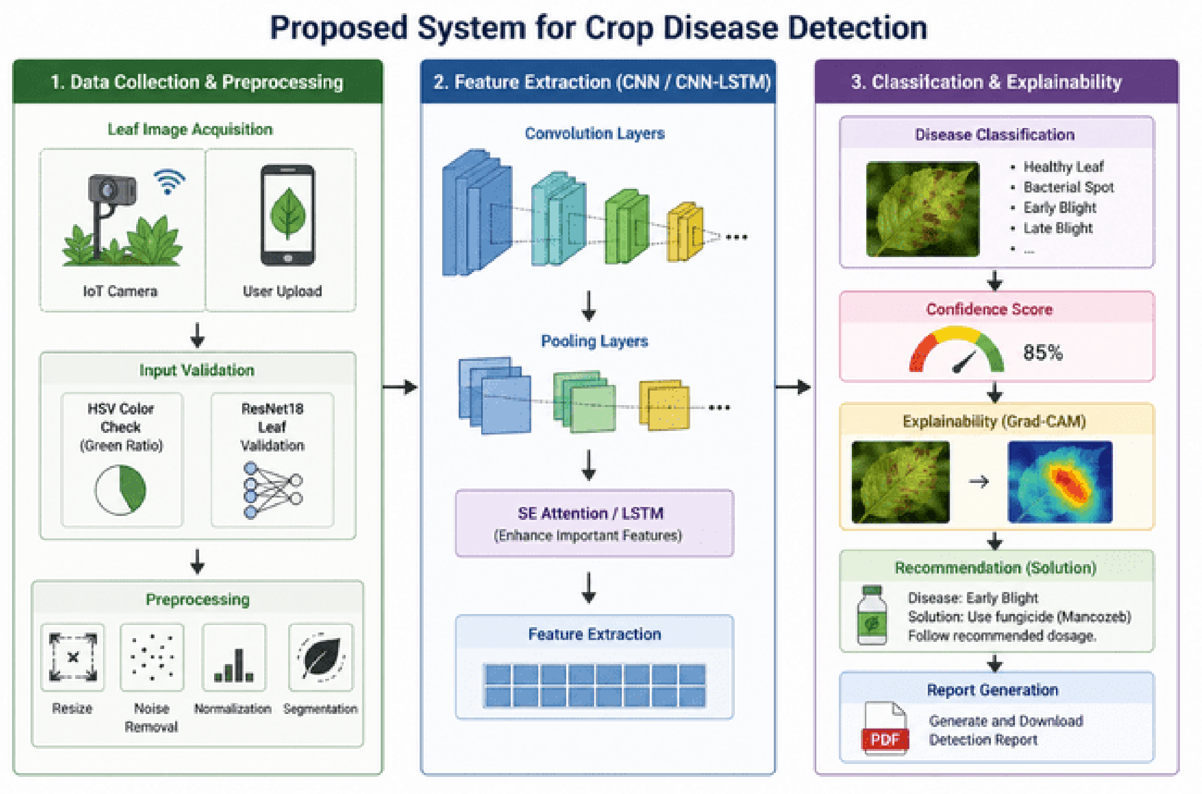

Fig. 2: Proposed System for Crop Disease Detection

The designed system architecture for crop diseases will be implemented in such a way that it will have three major processes. These include data acquisition and pre-processing, deep learning based feature extraction, and classification with explainability and recommendations. First, the leaves images are obtained from IoT enabled smart cameras that are mounted in the farms or by users who upload their images through their mobile or web application. However, to ensure that only legitimate data are used for analysis and processing, a validation process is carried out in two steps. The first step involves checking whether there are enough green pixels in the image using the HSV color space. If not, then the image will go through another validation process that uses a deep learning algorithm called ResNet18. Here, the image is classified into “Leaf” or “Not Leaf”.

Once validation of the image is done, the next step involves preprocessing the image to make it suitable for deep learning analysis. Preprocessing involves resizing the image into a uniform size, e.g., 224 × 224 pixels, removing noise from the image, normalizing pixels to range between 0 and 1, and extracting the leaf region by separating it from the background. All these steps are crucial for ensuring optimal performance of the deep learning model.

The second process involved in deep learning analysis is feature extraction. It involves the use of state-of-the-art deep learning models such as CNNs or CNN-LSTMs for analyzing images. CNN layers extract important spatial features including edges, textures, colors, and disease patterns in the leaf image. At the same time, pooling helps in reducing the dimensions of feature maps while still preserving most of the information contained in the feature map. On the other hand, techniques such as SE attention or LSTM can be employed to capture important spatial features of an image through the creation of relationships between different features in the leaf.

In the last phase, the process is capable of identifying and predicting diseases based on their classifications, and the model can produce an explainable result. The trained model produces the identified class of the disease, accompanied by the confidence level of the predicted output to determine its accuracy. To ensure the user understands the system and increases trust, the process uses Explainable AI algorithms such as Grad-CAM that produce heatmaps emphasizing the areas within the leaf where the prediction was made. Moreover, the process uses a recommendation engine that searches for the relevant treatment process using a predefined database that contains different treatment processes for various types of crops, including specific pesticides. Finally, the results produced by the process, including the predicted disease, confidence level, and explanation and recommendations, are presented in the form of a report in PDF format.

3.7 IoT Devices Used

Arduino Board

Fig. 3. The Arduino Board Used in the Proposed System

The Arduino board is an open source microcontroller board that is utilized extensively in IoT-based systems for data acquisition and device interface operations. The components present on an Arduino board include the microcontroller unit, digital and analog input/output ports, USB interface ports, and power supply unit. For the proposed system, the Arduino board has been chosen as a way to connect to sensors or other input devices to gather the real-time environmental data from the agricultural lands. It serves as a connecting link between hardware components and data processing system. The gathered data could either be processed by itself or forwarded to some other component like ESP32 for further processing.

ESP32 Device

The ESP32 is a highly advanced microcontroller that has been built for use in IoT applications. This microcontroller has built-in Wi-Fi and Bluetooth connections. The ESP32 contains high-speed processors, many GPIO pins, analog/digital interfaces, and communication modules (ESP-WROOM-32). The use of the ESP32 in this project ensures the effective real-time transfer of data from the agriculture farm field to the processing machine or cloud storage device. The ESP32 supports both wired and wireless communication; thus, it can be connected to sensors or cameras and help collect data wirelessly. The ESP32 provides enhanced performance and processing speeds compared to other microcontrollers

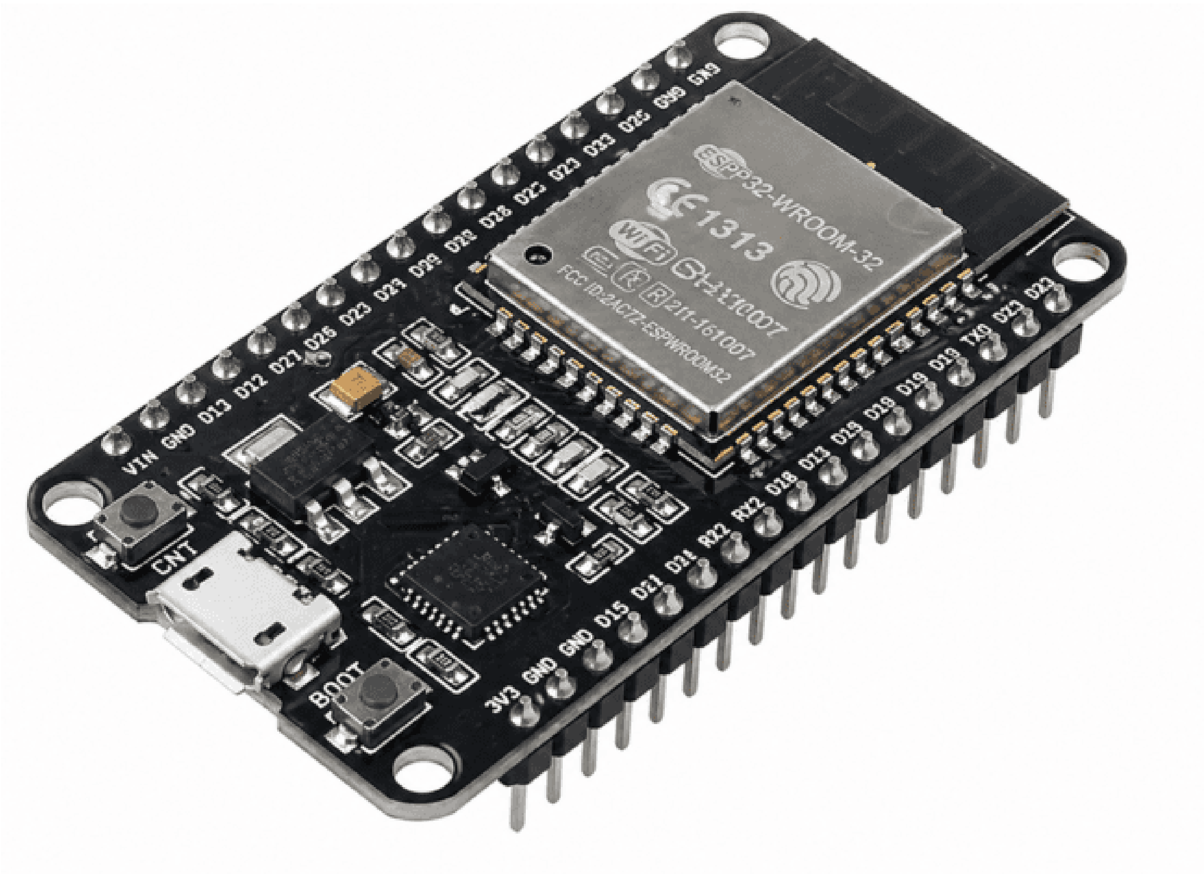

Fig. 4. ESP32 Module Employed in the Suggested Model

Algorithm: Crop Disease Detection System

Input: Image of Leaf (Either Captured Using IoT Device or Submitted by User)
Output: Disease Category (Healthy/Diseased), Probability, Heatmap, and Cure Suggestion

Step 1: Read the input image of leaf provided by the user or IoT device in the Flask back end environment and standardize it in an RGB image.
Step 2: Transform the RGB image into HSV format and apply a green mask in HSV space using the predefined thresholds to filter the plant part of the image. Find the ratio of green pixels and if this ratio is below 0.05, then discard the image.
Step 3: Resize the image to the size 224 x 224, transform it into the tensor and validate the image on the Resnet18 architecture. If the input image is not that of the leaf, then abort the process.
Step 4: Perform preprocessing, i.e., resizing the image to appropriate size, scaling the pixel values to [0, 1] range and expanding the dimensions as per the deep learning model requirements.
Step 5: Apply the CNN/LSTM model on the input preprocessed image to extract the spatial and sequential features and classify the disease type based on Softmax activation.
Step 6: Determine the predicted disease category using the maximum probability obtained in step 5 and mapping it to the relevant disease.
Step 7: Use Grad-CAM to produce heat maps showing the important parts of the leaf used in predicting the disease.
Step 8: Carry out canny edge detection and superimpose edges with the original image to add structure to the leaf.
Step 9: Access information on medicines for treating the identified disease from the database.
Step 10: Prepare the final output consisting of disease name, confidence of accuracy, heat maps and medicines for treatment. Offer the user the ability to save the result as a PDF file.

Mathematical Model for Crop Disease Detection

$$I \in \mathbb{R}^{H\times W\times C} \qquad (1)$$

$$F = f_{CNN}(I) \qquad (2)$$

$$S(i,j) = \sum_m \sum_n I(i-m, j-n)\, K(m,n) \qquad (3)$$

$$f(x) = \max(0,x) \qquad (4)$$

Squeeze-and-Excitation (SE)

$$z_c = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} U_c\,(i,j) \qquad (5)$$
$$s = \sigma(W_2 \cdot \delta(W_1 \cdot z)) \qquad (6)$$
$$\sigma(x) = \frac{1}{1+e^{-x}} \qquad (7)$$
$$\widetilde{U}_c = s_c \cdot U_c \qquad (8)$$

Bidirectional LSTM

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \qquad (9)$$
$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \qquad (10)$$
$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C) \qquad (11)$$
$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \qquad (12)$$
$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \qquad (13)$$
$$h_t = o_t \cdot \tanh(C_t) \qquad (14)$$
$$y_t = [\overrightarrow{h_t}, \overleftarrow{h_t}] \qquad (15)$$

Grad-CAM

$$\alpha_k^c = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A_{i,j}^k} \qquad (16)$$
$$L_{Grad-CAM}^c = \max\left(0, \sum_k \alpha_k^c A^k\right) \qquad (17)$$

Canny Edge Detection

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (18)$$
$$|\,G\,| = \sqrt{G_x^2 + G_y^2} \qquad (19)$$
$$\theta = \tan^{-1}\left(\frac{G_y}{G_x}\right) \qquad (20)$$

Image Fusion

$$Output(x,y) = \alpha I_1(x,y) + \beta I_2(x,y) + \gamma \qquad (21)$$

Classification

$$z = W_1 F + b_1 \qquad (22)$$
$$\hat{y} = Softmax(z) \qquad (23)$$
$$\hat{y}_{class} = \arg\max_k \hat{y}_k \qquad (24)$$

Loss Function

$$L = -\sum y_i \log(\hat{y}_i) \qquad (25)$$

Explanation of the Mathematical Model

Equation (1) represents the input leaf image.
Equations (2)–(4) describe Feature Extraction via Convolution Neural Network
Equations (5)–(8) represent the SE attention mechanism.
Equations (9)–(15) define the Processing via Bidirectional LSTM.
Equations (16)–(17) describe Grad-CAM visualization.
Equations (18)–(21) represent edge detection and image fusion.
Equations (22)–(24) define classification and prediction.
Equation (25) represents the loss function used for training.

Table 2:Hyper parameter of hybrid model

Hyperparameter	Value / Setting	Description
Input Image Size (IMG_SIZE)	224 x 224 x 3	The resolution all images are resized to before entering the network. 3 represents RGB channels.

Batch Size (BATCH_SIZE)	32	The number of images processed simultaneously during one training step.
Epochs (EPOCHS)	25	The maximum number of complete passes through the training dataset.
Learning Rate	0.0001 (10^{-4})	Controls how much the model's weights are updated in response to the estimated error each time the weights are updated.
Optimizer	Adam	Adaptive Moment Estimation. Computes individual adaptive learning rates for different parameters.
Loss Function	categorical_crossentropy	Used for multi-class classification tasks where labels are one-hot encoded.
Dropout Rate	0.4 (40%)	The fraction of input units randomly dropped to 0 during training to prevent overfitting.
Validation Split	0.2 (20%)	The percentage of the dataset reserved for validating the model's performance during training.
Early Stopping Patience	5	The training will stop early if the validation loss (val_loss) does not improve for 5 consecutive epochs.
CNN Filters (Both Models)	[32, 64, 128]	The number of output filters in the convolution operations for Layer 1, Layer 2, and Layer 3, respectively.
CNN Kernel Size	(3, 3)	The spatial dimensions of the filters used in the convolutional layers.
Pooling Size	(2, 2)	The window size used for Max Pooling to downsample the feature maps.
LSTM Units (cnnlstm.py)	64	The dimensionality of the output space for the Bidirectional LSTM layers.
SE Block Ratio (cnn.py)	16	The dimensionality reduction ratio used in the Squeeze-and-Excitation dense bottleneck layer.
Activation Functions	ReLU, Softmax, Sigmoid	ReLU for hidden layers, Sigmoid for SE attention weights, and Softmax for the final classification output.

The model takes an input image size of 224 x 224 x 3 to remain consistent in data processing. The batch size of 32 makes sure that the training process will be done efficiently and save memory at the same time. The model gets trained in 25 epochs with a learning rate of 0.0001 for obtaining better results. Using the Adam optimizer will provide faster convergence, whereas categorical cross-entropy is chosen as the loss function for multi-classification. Dropout rate of 0.4 and validation split of 20% are implemented for avoiding overfitting. Early stopping with a patience of 5 is also considered for preventing unnecessary training. Filters of 32, 64, and 128 with a kernel size of 3 x 3 and pooling size of 2 x 2 are considered for obtaining better features by using CNNs. Bidirectional LSTM consists of 64 units for handling sequential data. Meanwhile, the SE block ratio of 16 enhances feature selection.

4. Dataset

For this work, PlantVillage Dataset publicly available dataset was employed for the training and testing of the proposed crop disease detection system. The dataset is well known and is an open-source one, which is widely used by researchers for classification purposes in agriculture. It includes images of plant leaves in RGB format that belong to different crops and diseases. The images are preprocessed and are resized to be 224 × 224 pixels in size.
In the current study, the dataset used is a customized one with an emphasis on four crops, including Apple, Grape, Potato, and Tomato. The dataset contains 27,545 images in 21 different classes, with each class denoting a certain crop along with its diseases (healthy crops as well). All the images have been obtained under laboratory-controlled environments, with a leaf kept against a consistent background.

Dataset Link:
https://www.kaggle.com/datasets/emmarex/plantdisease

Table 3: Class-wise Dataset Distribution

Crop Name	Disease / Status	Image Count
Apple	Apple scab	630
Apple	Black rot	621
Apple	Cedar apple rust	275
Apple	Healthy	1,645
Grape	Black rot	1,180
Grape	Esca (Black Measles)	1,383

Grape	Leaf blight	1,076
Grape	Healthy	423
Potato	Early blight	1,000
Potato	Late blight	1,000
Potato	Healthy	152
Tomato	Bacterial spot	2,127
Tomato	Early blight	1,000
Tomato	Late blight	1,909
Tomato	Leaf Mold	952
Tomato	Septoria leaf spot	1,771
Tomato	Spider mites	1,676
Tomato	Target Spot	1,404
Tomato	Mosaic virus	373
Tomato	Yellow Leaf Curl Virus	5,357
Tomato	Healthy	1,591
Total	**21 Classes**	**27,545 Images**

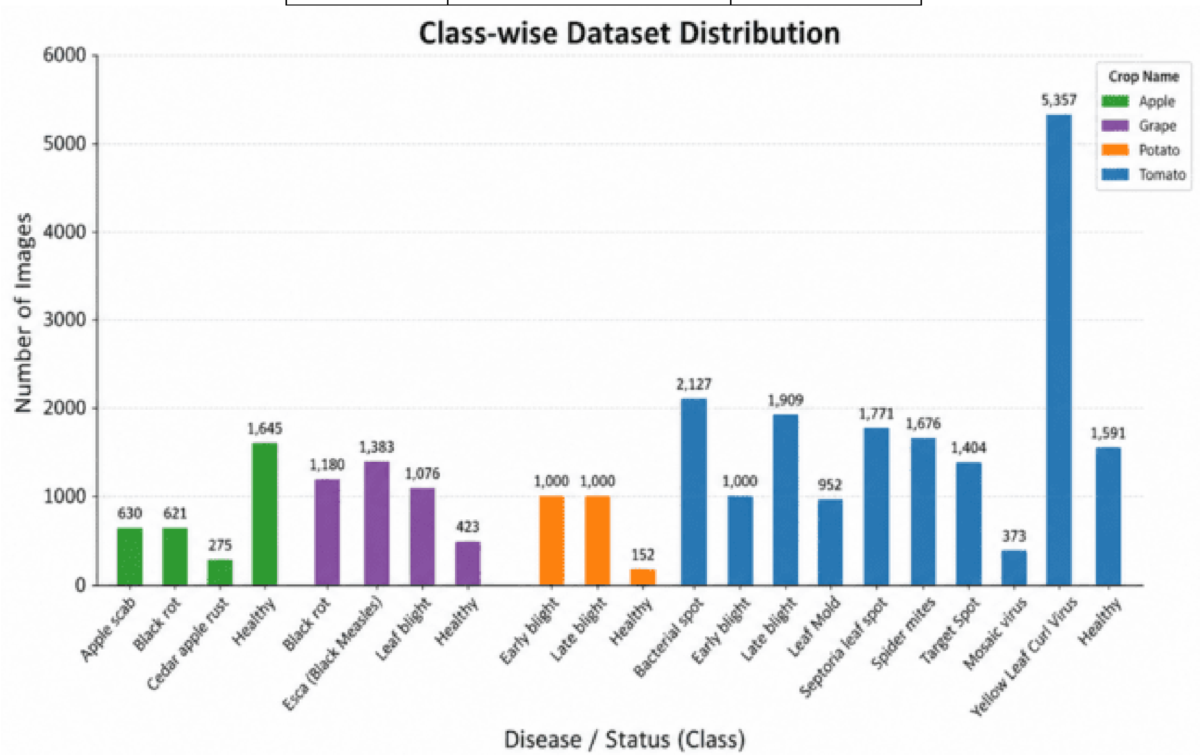

Fig. 5: Class-wise Distribution in Dataset

This dataset is made up of images from several crops that have various disease levels; healthy plants are also included. From the graph above, it is evident that some categories in the dataset have more images than others. For instance, Tomato Yellow Leaf Curl Virus category has more pictures than Healthy Potato category. The large difference in picture numbers poses a problem in training because there is a risk of bias due to the dominance of one category over another.

The controlled environment in which the dataset has been taken ensures that the model learns about the colors, shapes, textures, and other aspects of the leaves. Nevertheless, it may limit the model in the real-world application since the environments cannot be identical. Despite that, the dataset gives us a solid base for training models effectively.

b. Multiclass Classification Dataset

In case of multiclass classification, there are 21 classes in the dataset for each crop-disease combination. A particular class is for a particular type of disease. For instance, one class may be for early blight, another for late blight, etc. The difference in class distribution

makes clear that there is need to address imbalance. This is where methods such as dropout, data augmentation, and attention come handy for generalizing the model better.
The dataset facilitates effective learning for classification tasks, thus, ensuring success of the proposed methodology

5. Experimental Outcomes

Results are illustrated in this section using confusion matrices, accuracy graphs, and loss graphs. To give a thorough assessment of the performance of the framework, other performance measures like precision, recall, accuracy, and F1-score are also computed.

5.1 Disease Detection of Crops by Uploading the Images of Leaf

In the design of this proposed crop disease detection system, leaves' images are uploaded either using the user interface or using the Internet of Things (IoT) devices. In this process, leaves' images are first validated using the HSV color masking technique along with a ResNet18 algorithm in order to guarantee that only the image containing a valid leaf is processed further. This is followed by the processing of images to be done by resizing them to 224 × 224 resolution, normalization, and noise reduction.
Once this is completed, preprocessed images are then fed into the CNN architecture where the CNN extracts useful features like colors, patterns, and diseases in the images. Features extracted are then further improved by the SE mechanism, focusing on the parts of the leaf. In some instances, the CNN-Bidirectional LSTM model can also be used..
The extracted features then undergo full connection for classification into different categories of plant diseases. Furthermore, the application of Grad-CAM will also be applied to show the region affected by the disease in the leaf image. The results generated by the model will contain the disease predicted, confidence level, and suggested treatment.

5.2 Performance Metrics

In order to analyze the performance of the proposed classification model, other criteria are utilized along with accuracy since the imbalance of data might be encountered within the data set due to different numbers of images corresponding to each disease.
For this reason, Precision, Recall (or Sensitivity), Specificity, and F1-score performance measures are also calculated based on the confusion matrix.

- The value of Precision corresponds to the correct prediction of the positive cases.
- Recall (Sensitivity) shows how well the model can detect diseased cases.
- Specificity shows the capability of the model in detecting healthy instances.
- F1-score measures the precision and sensitivity simultaneously and balances both metrics.

Formulas for calculation of these performance criteria are provided below:
Precision = TP / (TP + FP)
Specificity = TN / (TN + FP)
Recall / Sensitivity = TP / (TP + FN)
Accuracy = (TP + TN) / (TP + TN + FP + FN)
F1 Score = 2 × (Precision × Recall) / (Precision + Recall)

Where:
TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative
The reason why F1 score is preferred over accuracy in the current project because of the imbalanced nature of the dataset. When we consider the case of multiple classes of disease, macro averaging and weighted averaging precision, recall, and F1 score are used. This would ensure that the proposed system works well and gives reliable results.

Fig. 6. Training vs Validation Loss

As can be seen from Figure 6, both training loss and validation loss vary throughout training periods. In the beginning, both training and validation losses are very high, implying that the model has yet to learn the data pattern and generalize to unseen data. However, throughout the training process, there is a consistent decrease in training and validation losses, meaning the model learns effectively and optimizes its parameters for higher accuracy. Another important aspect is the similar trend between the two lines, which means that no overfitting has occurred.

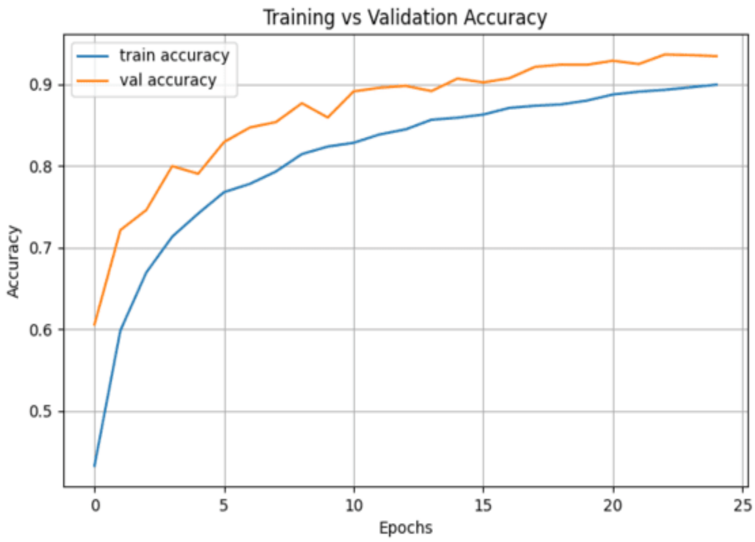

Fig. 7. Training vs Validation Accuracy

As is shown in Figure 7, this figure displays the training and validation accuracies for each epoch. At the beginning, the values of accuracies are relatively low because the system starts to learn from the data. However, the more epochs occur, the greater the training and validation accuracies become. In addition, the accuracy of validation becomes comparable with the training, which means that the model can generalize well without overfitting. Finally, the high accuracy at the end proves that the model can work effectively on classifying crop diseases.

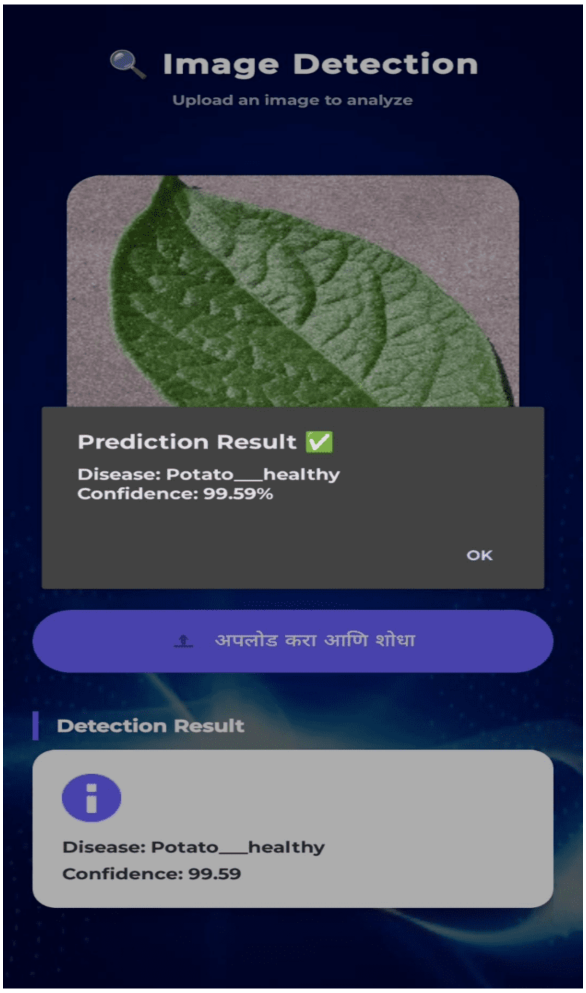

Fig. 8. Uploaded Image of Leaf and Predicted Outcome (Healthy Plant)

In Fig. 8, there is a screenshot of the prediction process using an image of a healthy plant leaf, showing how the user interface of the proposed system looks like. In the system, the image is processed to yield a predicted outcome showing that the plant is healthy (Potato healthy) with a confidence score of 99.59%.

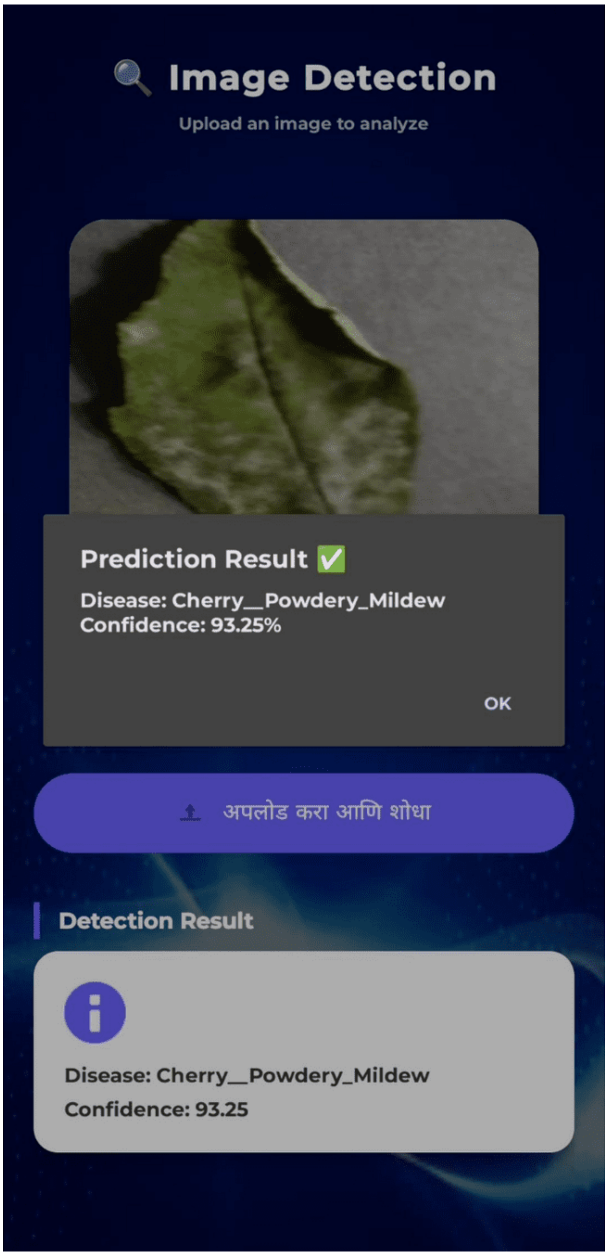

Fig. 9 Uploaded Image of Leaf and Prediction Output (Plant with Disease)

In Fig. 9, there is a screenshot showing how the uploaded image of a diseased plant leaf is processed using the prediction algorithm of the proposed system to yield a prediction outcome. The plant leaf in the image is correctly recognized by the proposed model as Cherry Powdery Mildew with a confidence score of 93.25%.

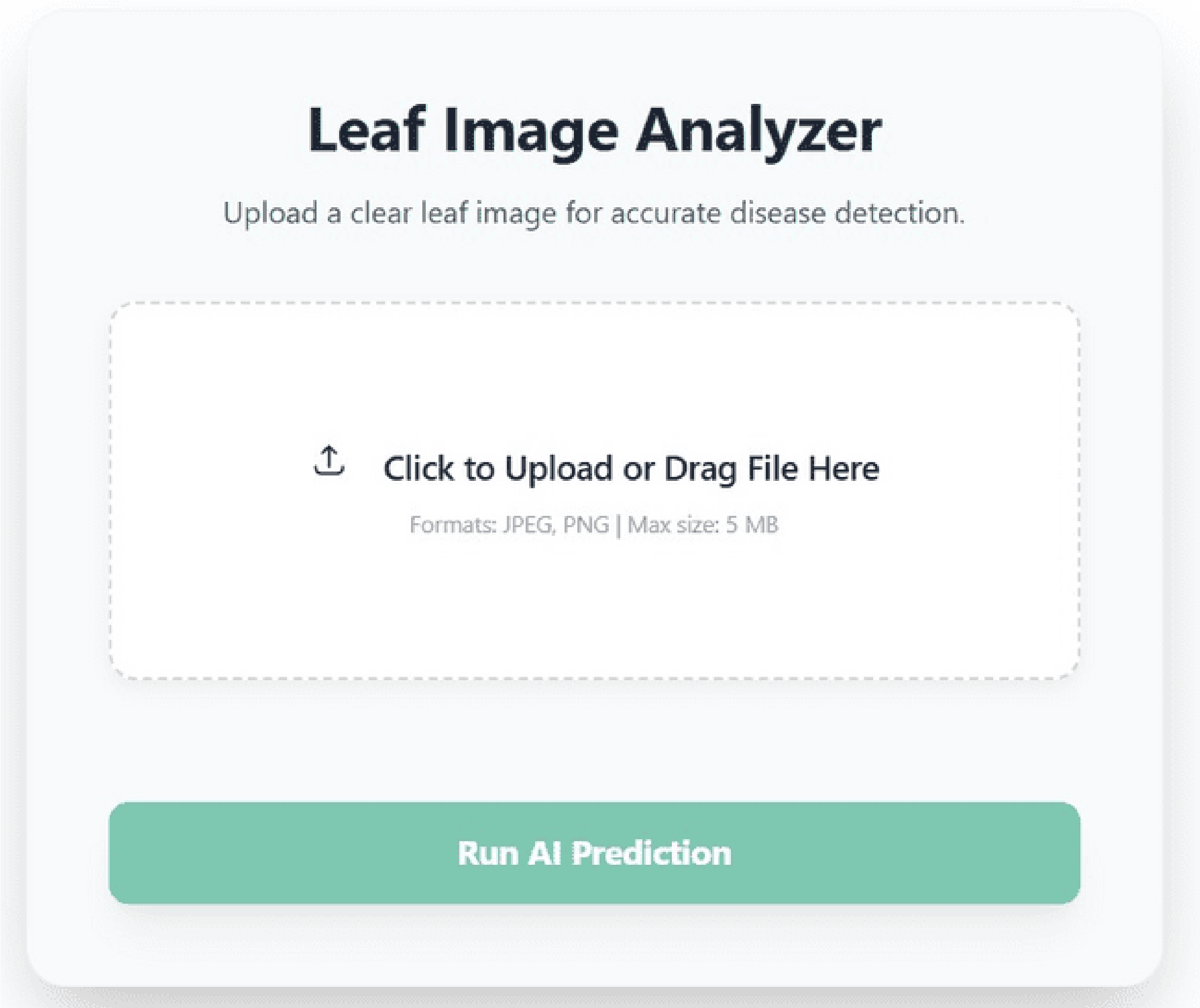

Fig. 10: Interface of Leaf Image Analyzer

Figure 10 shows the front-end interface of the proposed leaf image analyzer system, which is intended to offer an easy-to-use and intuitive user interface. Through the interface, the user can upload leaf images by either selecting the file or by dragging and dropping the image files into the application interface, allowing the use of the application by non-technical persons such as farmers. The supported image file formats include JPEG and PNG formats with a specified file size. After uploading the leaf image, the user can start the process of analyzing the image by pressing the "Run AI Prediction" button to launch the image analysis process.

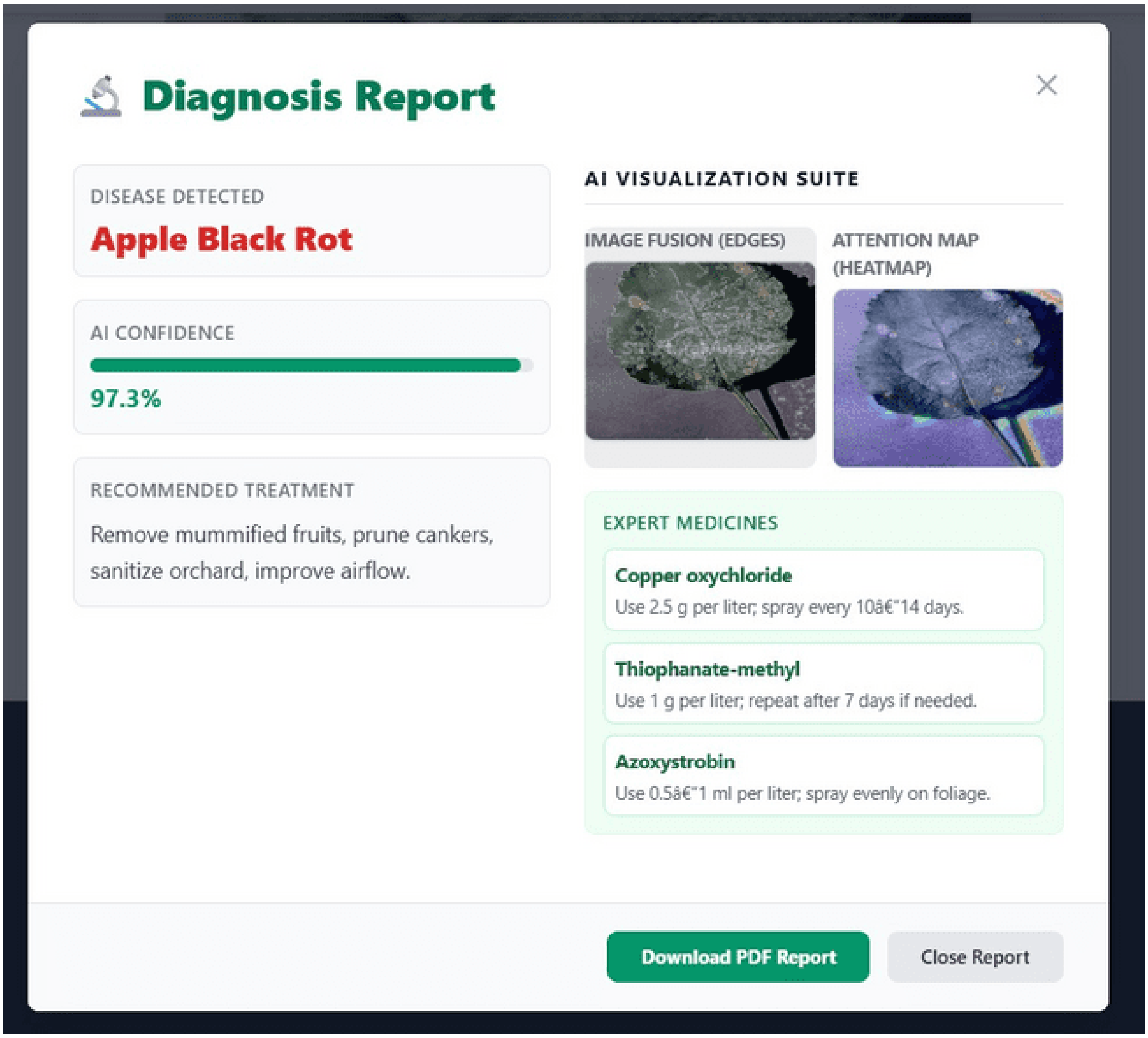

Fig. 11: Diagnosis Report with Disease Detection

The Figure below is an illustration of the diagnosis report created through the system for diseased leaves, where the disease detected was Apple Black Rot. There are several details included in the diagnosis report, such as the name of the disease, high confidence level (97.3%), and measures to be taken in treating the disease like trimming the disease spots and using appropriate fungicides. Aside from the text-based results provided, there is also a feature for AI Visualization, which contains two outputs: Image Fusion and Grad-CAM heatmaps. As can be observed, image fusion emphasizes the structure of the leaves (leaf veins and diseased spots), while the heatmap shows the parts responsible for making predictions. It also lists the recommended medicines according to experts, together with their proper application.

Fig. 12 Healthy Leaf Diagnosis Report

In Fig. 12, we see the diagnosis report of a healthy leaf that belongs to the Grape Healthy class with a high accuracy level of 99.7%. Unlike the diseased samples, where there are treatments required, the system verifies that the plant is not infected by any disease and only suggests general maintenance guidelines to ensure healthy plants. It can be observed that visualization results such as image fusion and attention heat maps are provided by the system, indicating that even in the case of healthy leaves, the model is able to focus on the necessary regions of the leaf. This uniformity in visual results enhances the transparency of the process. Moreover, avoiding unnecessary treatments ensures sustainable agricultural practices.

Novelty of Research

The uniqueness of this study is in the application of Internet of Things (IoT) concepts and image processing along with deep learning approaches for designing an effective real-time crop disease detection framework. Different from conventional systems based on manual identification or off-line image analysis approaches, this approach can monitor crops in real time through the use of smart IoT-based tools such as smart cameras to facilitate direct field collection of relevant data.
The uniqueness of the approach adopted by this study is also associated with image preprocessing procedures used to preprocess input data, which include removing noise, normalizing images, and performing segmentation tasks. It guarantees that the deep learning algorithm can receive high-quality input data, thereby contributing to increased effectiveness of disease recognition. In addition, this approach does not require manual feature selection because CNN-based algorithms can learn all critical features like color, texture, and diseases.
Moreover, the system takes into account such external parameters as lighting and background that are usually disregarded by other solutions based on training deep learning algorithms on artificial data sets.
At the same time, the combination of Internet-of-Things (IoT) technologies with deep learning allows delivering relevant information to farmers that will allow them to make decisions and undertake necessary actions on a preventive basis.The offered system is much more effective compared to the existing methods of smart farming

6. Conclusion

This research introduces an elaborate yet applicable system for detecting crop diseases based on the use of the Internet of Things (IoT), image processing and modern deep learning methods. This system manages to successfully automate the procedure of recognition of crop diseases based on images of plant leaves. The proposed method minimizes the need for manual inspections and expert knowledge through the automation of the procedure. By applying image processing techniques such as resizing, normalization and denoising, the system achieves improved quality of the data being fed to the algorithm, leading to increased prediction accuracy. The major advantage of this system consists in its ability to apply a combination of convolutional neural networks (CNN), squeeze-and-excitation (SE) attention mechanism and possible implementation of the CNN-Bi-directional long-short term memory network architecture. Such systems are capable of detecting color changes, textures and spots indicative of a particular disease. In addition to that, the application of SE allows focusing on important areas of the leaf and the LSTM component makes sure that the relationships between various areas are recognized. Finally, using Grad-CAM allows explaining the decisions made by the model..

From the experiments conducted, it can be seen that the model is able to provide high-level accuracy and stability as evident in training and validation graphs. The model also performs well under conditions of class imbalance because there are various methods of measuring its performance such as accuracy, precision, recall, and F1-score. In addition, the fact that the system provides real-time predictions and treatment suggestions is another added advantage.

While the data utilized are images of crops collected in a laboratory setting, the developed model exhibits great generalization ability. It can be used as a basis for developing a smart agriculture system that helps detect diseases at an early stage, minimize the loss of crop produce, and increase efficiency

Future Scope

In terms of future work, one may consider using lightweight and efficient models on edge devices or mobiles for detecting diseases offline and in real-time. Adding more variety to the image dataset by collecting images taken in different environmental settings (such as differences in lighting conditions, backgrounds, and occlusions) may lead to higher robustness. Advanced technologies like transformers for visual recognition or self-supervised learning may be used to achieve better results while having less labeled data available. The application could be developed further by including agronomic tips, such as dosage, optimal application time, and cost-effectiveness. Using IoT sensors for monitoring soil conditions (moisture, temperature, and humidity), it is possible to implement a full smart farming system.

References

[1] Z. Nyawose, et al., “Machine learning and deep learning methods for plant disease detection: A review,” Computers and Electronics in Agriculture, vol. 220, pp. 108-120, 2025. [2] V. S. Dhaka, et al., “Crop disease detection using deep learning: A review,” IEEE Access, vol. 9, pp. 160987–161011, 2021.

[3] M. Hasan, et al., “Deep learning approaches for plant disease detection and classification: A review,” Artificial Intelligence in Agriculture, vol. 5, pp. 1–14, 2020.

[4] E. A. Abade, et al., “A systematic review of convolutional neural networks for plant disease detection,” Computers and Electronics in Agriculture, vol. 176, pp. 105-117, 2020.

[5] S. Shelar, et al., “Plant disease detection using convolutional neural networks,” ITM Web of Conferences, vol. 44, pp. 03049, 2022.

[6] Y. Sun, et al., “EfficientNet-based deep learning model for plant disease detection,” Smart Agricultural Technology, vol. 2, pp. 100-112, 2022.

[7] I. Ahmad, et al., “Deep learning-based plant disease detection: A review,” Ecological Informatics, vol. 72, pp. 101-115, 2023.

[8] A. Upadhyay, et al., “Advances in deep learning for crop disease detection,” Artificial Intelligence Review, vol. 58, pp. 1–25, 2025.

[9] S. Hukkeri, et al., “CNN-based plant leaf disease classification,” The Open Agriculture Journal, vol. 18, pp. 1–12, 2024.

[10] W. Demilie, et al., “Comparison of machine learning and deep learning techniques for crop disease detection,” Journal of Big Data, vol. 11, pp. 45–60, 2024.

[11] B. Tugrul, et al., “Deep learning-based plant disease detection: A survey,” Agriculture, vol. 12, no. 8, pp. 1192, 2022.

[12] N. Afreen, et al., “Plant disease detection using Capsule Networks,” International Journal for Research in Applied Science and Engineering Technology, vol. 10, pp. 2345–2352, 2022. [13] P. Kumar, et al., “Plant disease detection using transfer learning and deep learning models,” Procedia Computer Science, vol. 218, pp. 123–130, 2023.

[14] R. Kumar, et al., “Optimized plant disease detection using CNN and metaheuristic algorithms,” International Journal of Computational En gineering Science, vol. 25, pp. 78–90, 2024.

[15] R. Sharma, et al., “Hybrid CNN-LSTM model for crop disease predic tion,” Computers and Electronics in Agriculture, vol. 225, pp. 108–118, 2025.

[16] S. Verma, et al., “Comparative analysis of CNN architectures for plant disease detection,” IEEE Access, vol. 13, pp. 45000–45012, 2025.

[17] D. Patel, et al., “Real-time plant disease detection using deep learning and mobile applications,” IEEE Transactions on Industrial Informatics, vol. 22, pp. 1120–1132, 2026.

[18] L. Zhang, et al., “A scalable deep learning framework for plant disease detection,” Scientific Reports, vol. 16, pp. 2100–2115, 2026.

[19] A. Singh, et al., “Hybrid machine learning and deep learning approaches for plant disease detection,” Environmental Monitoring and Assessment, vol. 196, pp. 300–315, 2024.

[20] K. Reddy, et al., “A systematic review of plant disease detection techniques using PRISMA methodology,” Indian Journal of Agricultural Research, vol. 58, pp. 220–230, 2024.

[21] N. Gupta, et al., “Deep learning models for crop disease detection and classification,” Computers and Electronics in Agriculture, vol. 224, pp. 107–119, 2025.

[22] X. Li, et al., “Enhanced CNN with attention mechanisms for plant disease detection,” Frontiers in Plant Science, vol. 16, pp. 1505857, 2025.

[23] Y. Chen, et al., “Region-based convolutional neural network for plant disease detection,” Journal of Plant Pathology, vol. 106, pp. 89–102, 2024.

[24] M. Khan, et al., “Multi-class plant disease classification using deep learning,” IEEE Access, vol. 13, pp. 55000–55012, 2025.

[25] P. Joshi, et al., “IoT-enabled smart agriculture system for crop disease detection using deep learning,” Sensors, vol. 25, no. 3, pp. 1450–1465, 2025